This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Previously Presented) A method of annotating an XML schema to provide information on how to parse a non- XML data stream containing delimiting portions and content portions to create an XML instance from the non-XML data stream, the method comprising:

including delimitation nodes within said XML schema that define delimiting characteristics of said non-XML data stream, where the delimitation nodes are child nodes in the XML schema, where instances of the content nodes in the XML instance are intended to contain the content portions of the non-XML data stream that are delineated by the delimiting portions of the non-XML data stream, the delimiting portions A identifiable by the delimiting characteristics of the delimitation nodes;

receiving said non-XML data stream;

parsing said non-XML data in accordance with information contained in said delimitation nodes of the XML schema by matching- the delimiting portion A of the non-XML data stream with the delimitation nodes to identify the content portions of the non-XML data stream, and generating the instances of the content nodes containing the identified content portions of the non-XML data stream, where instances of content nodes contain respective content portions identified by delimitation nodes of the corresponding content nodes in the XML schema; and

creating said XML instance by adding the generated content nodes to the XML instance, where the delimiting portions matched to the delimitation nodes are not included in the XML instance.

2. (Original) The method of claim 1, further comprising:

defining, in said XML schema, delimiters, TagNames, positional information within said non-XML data stream.

3. (Original) The method of claim 1, further comprising:

annotating said XML schema with said nodes in accordance with standard
conventions applicable to an XSD schema.

4. (Original) The method of claim 1, wherein said non-XML data comprises at least one of
flat files, EDI files, and COBOL files.

5. (Currently Amended) A computer-readable <u>storage</u> medium having stored thereon
computer executable instructions for defining an XML schema that describes a non-XML
data stream, comprising:

an instance tree view;

a text view of said XML schema; and

a properties view;

wherein to construct a schema, [[the]] <u>a</u> user creates a simple hierarchical tree
structure in the instance view and the properties associated with said non-XML data stream
are entered via said properties view to define characteristics of data contained within said
non-XML data stream<u>,</u>

<u>where the properties are expressed as delimitation nodes within said XML schema</u>
<u>that define delimiting characteristics of said non-XML data stream, where the delimitation</u>
<u>nodes are child nodes in the XML schema, where instances of content nodes in an XML</u>
<u>instance are intended to contain content portions of the non-XML data stream that are</u>
<u>delineated by delimiting portions of the non-XML data stream, where instances of content</u>
<u>nodes contain respective content portions identified by delimitation nodes of corresponding</u>
<u>content nodes in the XML schema, and where delimiting portions matched to delimitation</u>
<u>nodes are not included in the XML instance.</u>

6. (Previously Presented) The computer-readable medium of claim 5, wherein said text view
further comprises a color-coded view of a portion of said XML-schema associated with a
particular node.

7. (Previously Presented) The computer-readable medium of claim 5, wherein said specific properties comprise XSD properties.

8. (Previously Presented) The computer-readable medium of claim 5, having further computer executable instructions for maintaining an extensibility mechanism wherein said properties for specific nodes associated with differing types of non-XML data are specified.

9. (Previously Presented) The computer-readable medium of claim 8, having further computer executable instructions for maintaining a custom view for adding properties to each node that appear in said properties view.

10. (Previously Presented) The computer-readable medium of claim 5, having further computer executable instructions for maintaining a validation mechanism; and an instance generation mechanism.

11. (Previously Presented) The computer-readable medium of claim 10, wherein said schema is validated from a non-XML perspective.

12. (Previously Presented) The computer-readable medium of claim 10, wherein sample instance data is generated from said XML schema containing data from said non-XML data stream.

13. (Currently Amended) A method of providing schema editor extensions to a schema editor, comprising:

        providing assemblies containing implementation to extend a functionality of said schema editor;

        referencing definitions of interfaces for exposing extended functionalities to said schema editor; and

        adding custom properties to elements and attributes in the schema, the custom properties [[allow]] allowing the editor to define characteristics of the non-XML data within the XSD schemas in a standard fashion,

where the custom properties are expressed as delimitation nodes within an XML schema that define delimiting characteristics of said non-XML data, where the delimitation nodes are child nodes in the XML schema, where instances of content nodes in an XML instance are intended to contain content portions of the non-XML data that are delineated by delimiting portions of the non-XML data, where instances of content nodes contain respective content portions identified by delimitation nodes of corresponding content nodes in the XML schema, and where delimiting portions matched to delimitation nodes are not included in the XML instance.

14. (Previously Presented) The method of claim 13, wherein said extended functionalities include additional properties added to elements and attributes of the XML schema.

15. (Original) The method of claim 14, further comprising:

      providing a property manager that implements an interface to define custom properties for said elements and attributes.

16. (Original) The method of claim 15, further comprising:

      storing said custom properties within said XML schema.

17. (Original) The method of claim 15, further comprising:

      annotating said XML schema to describe non-XML data streams.

18. (Original) The method of claim 15, further comprising:

      providing a validator to validate said custom properties.

19. (Original) The method of claim 15, further comprising:

      providing an instance generator for generating an instance of said XML schema.

20. (Original) The method of claim 13, further comprising:

      annotating a schema being edited by said schema editor to include information about said extended functionalities.

21. (Previously Presented) A method of using information in an XML schema to parse a non-XML data stream to generate an XML document, the method comprising:

accessing the XML schema, wherein the XML schema comprises named elements that define elements that will contain content data in XML documents generated according to the XML schema, where some of the elements of the XML schema include one or more XML tags comprising delineation definitions; and

matching delineating portions of the non-XML data stream with the delineation definitions in the XML schema to identify portions of content in the non-XML data stream that are delineated by the delineating portions and adding to the XML document XML elements containing the identified portions of content, where the XML elements are instances of the named elements in the XML schema that include the delineation definitions, and where the delineating portions matched to the delineation definitions are not included in the XML document.